

4. SACWIS - A TECHNICAL VIEW

The SACWIS conceptual design provides a formal structure upon which the subsequent detailed design will be based. These design concepts have been developed to achieve the objectives and satisfy the information requirements discussed in previous chapters. In order to comprehensively evaluate the conceptual design, it is important to review the general criteria upon which the design is based.

4.1 SUMMARY OF THE DESIGN CRITERIA

Major criteria for the SACWIS design are as follows:

- As an information system, SACWIS must support functionally similar service delivery processes which occur within State child welfare sectors, for example:
 - the process of establishing client identity and opening a case record;
 - the process of selecting resources and services for delivery;
 - the process of monitoring client progress over time with respect to case goals, interim objectives, and services utilized; and
 - the process of monitoring client movement over time with respect to physical location and resources utilized.
- SACWIS must respond to varied information requirements of the different organizational entities for which it may be implemented. Specifically, SACWIS must be flexible with respect to the definition of a wide variety of client service activities and corresponding information content.
- SACWIS must include capabilities which, when properly used, promote the integrity and security of all information that is input, processed, and disseminated. Specifically, this means SACWIS must:
 - edit all input information and allow for its correction by the responsible data entry source before acceptance for further processing;
 - allow for logical separation of client specific identification data from highly sensitive case information; and

- allow for controlled access to SACWIS information for inquiry and reporting purposes by using inquiry source identification codes (e.g., agency, unit, worker, and password) which may be dynamically changed within the SACWIS environment.
- SACWIS is designed to address day-to-day operations and management needs with a secondary emphasis on research and evaluation. While not minimizing the importance of program monitoring and analysis, it is recognized that accurate and complete data for these needs are a by-product of a satisfactory response to child welfare service operations.

4.2 OVERVIEW OF THE SACWIS DESIGN CONCEPTS

The SACWIS design is founded upon several technical development concepts. It is important to review briefly the highlights of these concepts as they are the key to understanding the SACWIS design structure.

4.2.1 Components Form the Framework for SACWIS Design

The purpose of the SACWIS component definition is to provide a structural design framework which is functionally responsive to client service operations and reporting requirements. At the highest level, SACWIS component definition is based on functionally similar information requirements and client service processes identified within the States.

SACWIS consists of three separate applications: the Administrative, Child Welfare Services and Program Management Applications. The Administrative Application contains a single operational component that supports system administration. The Child Welfare Services Application contains seven functional components which provide information processing support of child welfare services operations. The seven components are Person/Family, Referral/Intake, Case, Placement, Court, Adoption, and Eligibility. The Program Management Application contains the Report Production and Interface components of the system.

Each of these components will be described later in this Chapter.

4.2.2 SACWIS is a Table-Driven System

The purpose of a table-oriented design is to provide flexibility with respect to system definition and maintenance. This approach minimizes the impact of system change because separately maintained tables may be updated rather than master files or computer programs. Within SACWIS, tables have been defined which anticipate the requirement for component adaptability to each user.

This approach will permit each SACWIS installation to define its own unique data classification and coding structure and maintain it over the life of the system. In doing so, the SACWIS system administrator will be responsible for the components selected in terms of information requirements, data code definitions, and output reports generated.

As a table-oriented system, SACWIS achieves a certain degree of "data independence" which simplifies system transfer. The major aspect of transferring SACWIS will be to define the contents of tables which are appropriate for a particular location.

4.2.3 SACWIS Employs Client/Server Computing Technology

The use of client/server computing is an integral part of the SACWIS design. It provides significant advantages over traditional computing and makes the SACWIS Prototype both more robust as a tool for case workers, and much easier to adapt to a variety of different State environments.

Client/Server Computing Defined

There are three primary levels in the client/server computing environment. They are the client, the server, and the network. There may exist multiple servers as well as multiple clients. Multiple networks may be employed to link servers at various sites. Typically the server accepts requests from the client for data services, performs those services, and returns data to the client. The client performs data manipulation with the data and presents the results to the user. This form of cooperative processing distributes the processing rather than the data.

The three major components of real-time computing are:

- presentation services;
- application logic; and
- data services.

In traditional computing all of the components reside with the mainframe or central computer. With client/server computing the components are distributed between the server (mainframe/central computer) and the client (typically a personal computer or workstation).

The move toward client/server computing has been evolutionary. Initially, with the growth of personnel computers came a desire to take advantage of improved computing tools at the PC level. This was accomplished by passing an entire file from a mainframe or file manager to the PC for manipulation. This had the disadvantage of allowing only one user to work on a file at any point in time. The second step in the evolution occurred when all of the presentation and application logic was moved to the client, and only data services were performed at the server. The third step created the contemporary client/server environment. Application logic is now commonly divided between the client and the server and some of the error checking previously associated with the data services performed at the server, now takes place at the client.

Benefits of Client/Server Computing

There are a number of benefits associated with client/server computing when compared with traditional computing. These include:

- savings in mainframe/central computer costs;
- applications are less costly to develop and maintain;
- increased control over individual personnel computers;
- increased transferability from use of more open systems; and
- increased end user productivity.

The relocation of presentation services, and some amount of the application logic from the server to the client, can result in a significant decrease in the cost of mainframe services. This is even more the case when compared with mainframe applications which attempt to provide a sophisticated set of presentation services to the user, all of which execute at the central computer.

Client/Server applications tend to be less costly to develop and maintain when compared to traditional computing applications. This is a result of both the ease of use of tools available to develop the client component of applications, and the separation of applications into a client (front end) and a server (back end).

Client/server environments also help to reestablish control over PC's distributed through out an organization. Tying these PC's together via the client/server network provides a vehicle for downloading software releases (version control) and managing the individual configuration and security profile associated with users and PC's.

Further, transferability is enhanced by the use of client/server computing. The client software provides a significant choice of hardware vendors without requiring changes to the application. Use of a relational data base product and the Structured Query Language (SQL) at the server can significantly reduce the cost associated with transferring an application from one hardware vendor environment to another.

Far and away the major benefit from client/server computing, and the reason for its inclusion in the design strategy for the SACWIS Prototype, is its' significant impact on end user productivity. Applications developed using the sophisticated user interfaces available for the Client/Server environment, especially if they are integrated with office automation tools, can provide a whole new way for end users to interact with computers and accomplish their work.

With traditional computing, child welfare services applications were developed primarily as management information tools. The high cost of providing direct access to computers for social work and other staff, coupled with limitations on the tools available to manipulate information at the desktop, tended to encourage applications driven by forms completed by social workers and entered by clerical staff. In these types of applications all of the information gathered came

at the expense of the caseworker's time to provide services. It was therefore necessary to balance the information needs of the agency with the cost in social worker time required to obtain the information and enter it onto forms. There was little payoff to the case worker in terms of automation because most of the forms they were required to fill out still needed to be done manually even if some or all of the information resided on the computer.

Client/server computing, with its' associated workstation (PC) at the desktop level, opens up the possibility of an entirely new relationship between the social worker and their computer, with additional positive implications for the availability of program management information for planners. The workstation can now integrate child welfare services applications with an expanded set of application and office automation tools to provide the worker with ease of entry for case information, as well as access to a variety of ways to organize and view information. Though the workers role expands with the additional responsibility of entering and maintaining information about clients and cases, the trade off is that the information can be entered directly into the computer instead of recorded in the case folder and then entered onto forms for data entry. Additionally, once the information is in the system it is no longer necessary to manually fill out various forms with that same information, as it can be populated automatically from the previously stored information, and produced on demand via local printers.

The benefit for program planners of this new type of environment is obvious. A much greater quantity of data is available, and that data is of significantly higher quality than is traditionally found in the mainframe/terminal/centralized data entry environment. This benefit is achieved without incurring the cost in social worker time to fill out multiple, redundant forms. This greater volume of data provides ways to monitor compliance with good practice and regulations, recognize training issues, and evaluate strategies and resource allocations based upon outcomes.

4.2.4 SACWIS Client Environment and Construction Tools

The SACWIS Prototype requires a robust client environment. A construction tool had to be chosen to develop the application. It had to be comprehensive, flexible, inexpensive, and with a good chance of being supported by its vendor for years into the future.

Client Environment

The client environment consists primarily of the type of workstation or PC and its operating system. The major client environment choices today consist of Apple Macintosh computers with the System 7 Operating System, IBM compatible computers running either Windows or OS/2, and workstations running UNIX. The IBM compatible/Windows environment has been chosen for the SACWIS model. The model will also operate with Windows running under OS/2. This environment represents the largest segment of the current market place for microprocessors and operating systems.

Client Construction Tool

The two most popular construction tools for Windows based client applications are Microsoft's Visual Basic and Powersoft's Power Builder. Other tools are available, some for generic Windows environments (Visual Basic, Power Builder, etc.), and some for vendor specific environments. Some vendor specific products have very strong enterprise wide capabilities, allowing integration with other tools available from that vendor to create a strong development environment. These vendor specific products, however, limit the transferability of the model. They also create more of a problem in locating trained programmers familiar with the products, and are more problematic in terms of assured ongoing support from the vendor.

Both the Visual Basic and Power Builder products are extensively used for the development of client/server applications at the current time. Both products provide a comprehensive environment for the development of Windows based applications. Both can be used with a variety of middleware products to connect to networks and access data services from servers. Both have a large enough base of users that it seems reasonable to assume that they will continue to be supported and enhanced in the future.

Visual Basic has been chosen as the client development tool for the SACWIS model. The decision is based upon a somewhat larger user base, a larger set of programming objects available

to create the user interface, and our conclusion that Microsoft has made Visual Basic key to its other software products, ensuring a significant continued development investment in the product. Visual Basic also provides easy interface to the Access data base engine used at the prototype level of the SACWIS design.

4.2.5 Server Utilization

The server can consist of one or more platforms at which data services for clients reside. This might mean a server resident on a local area network and/or a centrally located mainframe computer. Software will be required on the server to facilitate communication via the network with the clients.

The data services provided by the server include access to data, some validation, and enforcement of business rules through data formalization of the organization's entity relationships. Some of the application logic may also reside at the server.

There are two primary methods by which clients communicate with servers in client/server computing. These methods are: remote procedure calls (RPC's) and a strict relational interface where the data base commands (almost always SQL) are passed directly to the server where they are executed and the results returned to the client.

Remote Procedure Calls

The reasons to place application logic at the server and use RPCs involve performance and data integrity. Performance is enhanced because the data access commands have been previously optimized and require less resources. Also intermediate data is not returned to the client to be examined prior to initiating additional calls to the data base, therefore, only final results from a procedure are returned. Data integrity is enhanced by the use of RPC's in that complex transactions involving multiple calls to the data base are completely contained in the remote procedure so that the changes to the data base may be committed or rolled back depending on whether or not errors occurred in the processing.

RPC's are extensively used in client/server system development, however, they are specific to the hardware and middleware in use at the server level and limit the ability of a particular system

to be transferred to different operating environments. Middleware issues are also more complex when utilizing RPC's than when standard SQL strings comprise the communication between the client and server.

SQL Calls

Some of the relative disadvantages of SQL Calls have been mentioned above in showing the advantages of RPC's. There are also advantages to the SQL call approach, the most important being that the SQL call model (relational interface) may be more easily moved between platforms and operating environments without the extensive re-writes often associated with RPC's. Use of this approach entails close attention to the definition of what transactions constitute a unit of work. The decision as to which approach to use or more commonly, what mixture of these approaches is most appropriate, is best made at the implementing site where considerations regarding network availability and speed, server size and degree of utilization and risk tolerance can be weighted. The SACWIS prototype development team is currently undertaking a major analysis of these approaches in order to determine the most efficient and practicable developmental methodology.

4.2.6 SACWIS Development Platform(s)

SACWIS model prototype development will be conducted on a Local Area Network (LAN) using the ACCESS data base as if the system was located on a single computer. Version control software will allow several technical staff to be active in the development at the same time. A move to a true client /server environment may be undertaken at a later date. Such a move would entail the use of commercially available utilities to convert the exiting database requests to operate in a targeted client/server environment. At the present time both the client and the server are the same computer. The data services are separated from the other processes so that they may be moved easily to another server when the model is implemented. The model currently assumes that some level of RPCs will be used to communicate with the server. Units of work at the client (work done by the user at the PC which must be saved or discarded before moving on) are designed to be small and self contained to lend themselves more easily to RPCs, if and when the decision is made as to whether the RPC approach will be implemented.

Almost all client server applications are now being designed with data stored in RDBMS format. This is a necessity if a straight relational interface is used for the interface between the client and the server and may or may not be necessary if an RPC interface is used. This is dependent upon the server platform and the middleware selected. A significant advantage of RDBMS is the uniformity that the SQL method of accessing data provides across different vendors. This type of open solution enhances transferability, and reduces modification and maintenance costs.

The SACWIS model is being designed using RDBMS and SQL technology. This means that the prototype can be developed on a single platform (PC) using Microsoft's' ACCESS relational data base. During implementation of the model the data base is then transferred to a server using another relational data base product without changes to the data base structure for other than site specific requirements. The SQL for the prototype can either be left as is at the client for the straight relational interface, or moved to the server to be accessed via RPCs.

4.2.7 Connectivity/Middleware

For client/server computing to work, software is required to facilitate the passing of data back and forth between the clients and the server over the network. Various type of networks may be employed in the client/server environment, and their choice is usually site and vendor platform specific. The middleware which handles the data communications has a client and a server component. It is the communication between these two components which actually allows clients and servers to 'speak' to one another. There are a number of products by various vendors that provide this middleware function. The selection of a specific product may be driven by the vendor specific server platform and network environment available in the implementing site, though many of these products support for multiple server platforms. Three of the available products are **Open Client/Open Server** from SYBASE, Oracle's **SQL NET** and **SQL CONNECT** and **Encina** from Transarc Corp. Some of the middleware solutions support both the relational interface and the RPC interface between clients and servers, but many do not.

The SACWIS design can accommodate any middleware site specific choice. As discussed previously, the prototype is being developed on a single platform acting as both client and server. As a result no middleware is currently required. At implementation, when the data base and data services are moved to a specified server platform, middleware will be required on client and

server platforms. If a straight relational interface is employed, code will be required at the client application (modification to Visual Basic application) to pass the SQL commands to the network and to access the data as the results are returned to the PC. If an RPC interface is to be implemented, additional code will have to be written at the server to initiate the SQL commands moved from the PC, and to package the results for communication back to the client.

4.2.8 User Interface Paradigm

Much of the user interface paradigm for the SACWIS model is set by the selection of the Windows graphical user interface (GUI). The use of Visual Basic as the construction tool provides a rich environment of windows objects for displaying and entering data, as well as controlling navigation, providing messages to users, and offering general and context specific help.

The design for the navigation of the SACWIS model involves the use of icons to represent the major entities and data groupings, with folders (groups of screens) which are accessed by clicking on tabs on the screen. This permits navigation between entities and groups of data about entities, with a minimum of hierarchy. Users are able to go in a direct manner (lateral navigation) to the information they want without moving up and down hierarchical chains of screens.

The SACWIS model design is client based rather than case based. The primary entities in the system such as persons, families, courts, foster homes, etc., exist in the system one time, and are linked together in many to many relationships. As a result the client has a history of service episodes (cases) rather than occurring in the system multiple times when new cases are opened for the same individuals.

The model will allow a user to navigate easily between major entities via the relationship between them. For example, a user can select a client from their client list and display client summary information. Included on the summary information is a list of all reports of child maltreatment (referrals) in which the client has been included. From that list the user can click on one of the referrals to navigate to the referral module with a display for the selected referral. The summary for that referral will show all clients associated with the referral. The user can then click on a different client placing them back at the client module but with a display for the newly selected

client. This same type of navigation also allows the user to move freely between clients and families, placement facilities, and courts.

4.3 OVERVIEW OF THE SACWIS APPLICATION

Exhibit 4.1, Functional Requirements Cross Reference Table, shows how the major applications in SACWIS support the functional requirements mandated by the SACWIS Action Transmittal.

As previously mentioned, the SACWIS design concept consists of three separate applications. The Administrative Application contains the various tables and organizational views which support the Child Welfare Services Application. The Child Welfare Services Application addresses the needs of the caseworker who provides child welfare services. The Program Management Application supports the report production and interface components of the system.

The SACWIS design relies upon decision-tree and event technology to capture and summarize information, as well as to provide alerts (ticklers) to workers regarding necessary activities to be completed. Event data can be accessed through the main entities, such as Person, Family, Referral, and Resource, or through specialized search capabilities related to type of event, date of event, etc.

The Administrative Application is accessed much less frequently than the Child Welfare Services or Program Management Applications, and is separated for both performance and security reasons.

4.3.1 Administrative Application

This component of the system provides the functionality to establish allowable values for attributes or elements; define events and ticklers; establish lists of courts and attorneys; and define caseloads, staff, units, and other organizational entities of the State agency. Through this application, Persons,

Families and Referrals can be assigned to one or more caseloads (primary and secondary workers). Assignments are made to a caseload rather than to an individual so that a history of specific staff persons may be tracked, over time, for a particular caseload or file. This structure also allows a caseload to be shown as uncovered at a specific point in time.

In the assignment process, staff specialties and caseload functions can be used to ensure a more appropriate assignment. For example, a caseworker or staff member may have specific language capabilities or skills that are of particular interest. Or a caseload may have specific associated functions such as emergency response investigations or medical needs.

Caseloads are assigned to units in a fashion similar to caseworkers being assigned to caseloads. Caseloads may be moved from unit to unit, and a history of this movement is created in the system. Supervisors are assigned to a unit via a special supervisors' caseload which provides inquiry, approval, and update functions for all other caseloads in the unit. Units are assigned to organizational entities, that in turn are assigned to higher level organizational entities. This open structure allows the system to accommodate varying organizational structures such as sub-offices, offices, regions, counties, etc., for reporting purposes.

Staff have specific actions and forms that are useful to them through this application. These include:

- Caseload List
- Tickler List
- Staff Profile
- Person Search
- Family Search
- Referral Search
- Resource Search
- Value Table

- Event Table
- Tickler Table
- Staff Maintenance
- Unit Maintenance
- Caseload Maintenance
- Organization Maintenance

4.3.2 Child Welfare Services Application

The Child Welfare Services Application consists of several components. Each of the components consist of specific entities related to child welfare services.

Person/Family Component

The Person/Family component includes the primary entities of Person and Family. A Person may be a member of multiple Families, with a different relationship to each. Cases can be tracked by Person or Family. Person and Family may have several identifying numbers some of which serve as cross-references to other systems. Person and Family can have Events and Ticklers associated with them. Events are activities entered by staff or created automatically by the system. Ticklers are automatically created and automatically deleted when the action that satisfies the Tickler is entered. Collateral Persons are also entered into the system and can be associated with Persons or Families. Such Persons can be referenced in service plans and contact recording.

Examples of actions that a worker may perform in the Person/Family Component include:

- Obtain Person Summary
- Obtain Placement History
- Obtain Status of Case
- Maintain Person Names
- Maintain Core Person Data
- Maintain Client Address and Phone Number

- Maintain Family Associations
- Maintain Placements
- Maintain Person Caseload Assignments
- Maintain Court Associations
- Maintain Person Case Status (including closure)
- Maintain Person Events
- Obtain Family Summary
- Obtain Family Case
- Obtain Family Collaterals
- Maintain Family Names
- Maintain Core Family Data
- Maintain Family Address and Phone
- Maintain Family Caseload Assignments
- Maintain Family Case Status (including closure)
- Maintain Family Case Plan
- Maintain Family Case Events

Referral/Intake Component

The Referral/Intake component involves the primary entity of Referral. A Referral or report of abuse or neglect may have multiple associated Persons (child victims and adult perpetrators). Conversely, a Person may be involved in multiple Referrals. A Referral can have a single reporter and multiple allegations. Allegations may be disposed of individually. The system supports the recording of multiple cross reports (notifications) to law enforcement and other agencies, and tracks Referral activities from screening through assessments or investigations as well as dispositions, through Events and Ticklers.

Some of the activities that workers can access through this component are:

- Obtain Referral Summary
- Obtain Narrative Summary
- Obtain Reporter Information
- Maintain Referral Information
- Maintain Reporter Information
- Maintain Allegation Information
- Maintain Referral Caseload Assignments
- Maintain Cross-Reports Information
- Maintain Referral Events

Case Component

The Case component provides case management and tracking functions for Person and/or Family cases. Case administration involves tracking individual service episodes and the service status of cases during the service episode. Case planning is supported by the system by setting a service goal, planning services, tracking services delivered, and periodic reevaluation and modification of case plans. Activities are tracked through Events and Ticklers.

Basic actions of the caseworker will include:

- Maintain Case Component Information
- Maintain Case Plans
- Maintain Person and Family Assessments
- Maintain Service Episodes
- Maintain Case Events

Placement Component

The Placement component involves the primary entity of Placement Facility. A Placement Facility is the actual physical location at which a child is placed. This facility can be a foster home, group home, adoptive home, etc. If a Placement Facility is part of a group home or foster home agency, the information about the agency is maintained in the Placement Organization entity. Information about foster parents, group home staff, or adoptive parents is maintained in the Placement Provider entity. Information specific to adoptive parents, which may be used for matching adoptive parent applicants with children available for adoption is maintained in the Provider Adoption entity. The Placement Category entity provides a way to classify the total available bed spaces in a facility by individual bed, bedroom, cottage, etc. This provides maximum flexibility in describing available space for the purpose of automated vacancy matching by the system. Events and Ticklers track activities associated with the Placement Facility, Providers, or Organizations.

Worker activities supported by this component include:

- Obtain Placement Summary
- Obtain Placement Census
- Obtain Placement Organization
- Obtain Placement Events
- Maintain Placement Facility
- Maintain Placement Categories
- Maintain Placement Organization
- Maintain Placement Providers
- Maintain Adoption Applicants
- Maintain Placement Events

Court Component

The Court component involves the primary entities of Court, Hearing and Attorney. This component will track and support the involvement of the Court with Persons. Hearings and outcomes are recorded. Attorneys are associated with Persons and specific Hearings. Legal status information is maintained in the system.

Worker activities supported by the design are:

- Obtain Court Summary
- Obtain hearing Summary
- Maintain Hearing Information

Adoption Component

The Adoption component provides the capability to track the status of adoptive parent applicants; create potential matches between adoptive parent applicants and children available for adoption; and track the activities associated with the adoption process.

Worker activities included in the design are:

- Obtain Provider Adoption Data
- Maintain Adoption Information

Eligibility Component

The Eligibility component assists in the determination of Person IV-E eligibility for Foster Care. A history of eligibility and the redetermination process is supported.

The primary activity supported by the system is:

- Maintain IV-E Eligibility Information

4.3.3 Program Management Application

Program Management is the component of the system that provides for the selection, formatting, and printing or displaying of reports, in support of short-term and long-term planning, monitoring and evaluation. It is also the component of the system that provides for linkage with other systems so as to receive and transmit case and client information.

Report Process

The intent of the Program Management report process is to offer an automated framework within which the user has the capability to define and produce a variety of statistical reports for program management needs, a series of alerts or "ticklers" for caseworkers and other users, and several standard reports which meet specific reporting requirements.

While the budget of the project prohibits the development of a significant number of reports, the Project Team will develop samples of reports for three tiers of program management reports.

They are:

- Program Activity Reports - related to day-to-day operations and exceptions, (e.g., referrals, petitions, open/closed cases, placements, services provided, staff allocation, etc.);
- Quality Assurance Reports - related to data accuracy and case audits, compliance and evaluation; and
- Program Planning and Evaluation Reports - related to long-range planning and longitudinal monitoring.

Other examples of informational products that may be of use and/or interest to the various levels of child welfare services staff are the following:

Case Managers

- Time sensitive alerts/tickers for upcoming and past due case activities
- Client/Family/Referral Case Load listings
- Automatically populated form templates
- Notifications to designated individuals, organizations and other entities
- Data obtained from interfaces with other automated systems
- Historical summaries of past involvements

Supervisors

- Parameter driven alert/tickler notifications of past due or critical activities
- Caseload listings
- Case Worker assignments by Client/Family/Referral
- Staff on-call status
- Analysis of past due activities by staff, case type and other parameters
- On-line access to case information, including narratives

Managers/Administrators

- Caseload analysis by unit, geographical area, type, etc..
- Summaries of activities completed, due, past due
- Time line analyses
- Historical comparisons of designated indicators
- Demographic profiles of clients
- Staffing patterns

Research

- Service outcomes by client demographics
- Historical comparisons of designated activities
- Unmet needs analyses
- Client profile shifts over time

In the development of the report samples, the Project Team will make use of statistical interface and report generator packages, as they are generally available within specific user environments, so as to demonstrate the utility and practicality of this approach.

The sample report production will also include data verification and extract capabilities necessary to the satisfaction of AFCARS and NCANDS reporting requirements.

Interface Process

While the SACWIS Interim Final Rule requires that SACWIS provide for timely and automated exchange of information between SACWIS and the following systems: title IV-A (AFDC), title IV-D (child support), title XIX (Medicaid) and NCANDS; there are numerous issues that prevent the Prototype from approaching this requirement with anything more than data import/export capabilities. Following is a brief discussion of the issues.

AFDC, child support and Medicaid information systems are often implemented on different and incompatible platforms using distinct, incompatible communications networks. In addition, most States have implemented these entitlement applications on mainframe systems as distinct, discreet databases rather than implementing a comprehensive, integrated database system where data elements have consistent meaning across all of these systems. The three systems often use different database management systems on separate mainframe systems with periodic exchange of data between systems using interfaces which translate data from the format and context of one system, for example FAMIS (IV-A) to the appropriate format and context of the target system, for example Child Support Enforcement (IV-D). Both systems must maintain reference data for the other system in order to enhance interface efficiency, for example the title IV-A Member ID or the title IV-D Case ID.

Besides the technical impediments to creating effective SACWIS interfaces, various States have limited resources available to implement these interfaces. Information technology organizations have relatively small staffs and their knowledge of, and skill in, the emerging client/server technologies is often limited. Furthermore, many States have limited computing resources and not provide the required additional capacity for a robust SACWIS interface. Finally, the entitlement systems listed above provide similar data, but often define data nuances that differ by State and by system.

While there are a number of issues to be solved, the interface requirements can be addressed by States in a couple of different ways. They are:

- | | |
|------------------------|---|
| <i>On-line:</i> | real-time data transfer between SACWIS and FAMIS (IV-A), Child Support Enforcement (IV-D), Medicaid (Title XIX), etc. |
| <i>Batch:</i> | periodic, normally overnight, transfer via tape or electronic batch file transfer between SACWIS and FAMIS (IV-A), Child Support Enforcement (IV-D), Medicaid (Title XIX), etc. |
| <i>Near Real-time:</i> | periodic electronic transfer of information between SACWIS and FAMIS (IV-A), Child Support Enforcement (IV-D), Medicaid (Title XIX), etc. |

The advantages and disadvantages of each of these approaches are as follows:

On-line/Interactive Interfaces

While on-line, interactive, real-time data transfer between SACWIS and the mandatory interface systems is desirable, several factors make this option both complex and cost prohibitive:

1. SACWIS and the mandatory systems to be interfaced with are often implemented on diverse platforms (mainframes, minicomputers, workstation servers, and databases) from different vendors. The possibility exists for SACWIS to provide a cost effective, terminal server gateway to the target systems, e.g. 3270 terminal emulation to IBM and IBM-compatible mainframe FAMIS, Medicaid, or Child Support Enforcement systems or terminal emulation to non-IBM platforms using standard DEC VT100 terminal access. This interface would not require system modifications to the desired

system, however, SACWIS users trying to navigate the different systems may not find this easy task. Training, performance, and system nuances may make this an unacceptable alternative in some states.

2. System to system, real-time access via an application program interface would be complex and require modification to the target system. Depending upon the application platform, different pseudo-standards would be required, for example LU 6.2 (APPC) on IBM platforms and Remote Procedure Calls in a TCP/IP environment. In any case, this interface requires modification of the system SACWIS is attempting to transfer data (from or to). In some cases, SACWIS would require two or three different interfaces. Many State information technology staffs do not have the staff support to develop, maintain, or support this real-time access interface and may not have the flexibility, or pay structure, to recruit these staff. However, from an end-user perspective, this would be the desirable alternative.

One consideration that states must realize with the implementation of real-time SACWIS interfaces is that the interfaces must allow for the unlikely event that communication between the systems may be disrupted and some form of backup queuing must be available should the interface become inoperative for some period of time.

Batch Interfaces:

Currently most states have implemented comprehensive batch interfaces between their title IV-A, title IV-D, Medicaid title XIX), and child welfare services systems. These interfaces include both file transfer between systems and/or the exchange of tapes. Often, these data transfers occur nightly as part of the batch cycle. Generally, each system implements a generalized interface program that does the following: translates from one computer system numbering system to the target system format; provides data edit checks; makes use of control fields to determine processing for interface flexibility; enters interface data into an in-process queue if appropriate; and provides a focal point for exchanging information via an interstate network (for example the Child Support Enforcement Network (CSENET)).

The existing interfaces provide the vehicle for the exchange of information among the systems. Each system user determines whether they consider the interface data to be the best source of information and updates the target system, or discards the interface information.

Near Real-time Interfaces:

These interfaces are electronic, batch interfaces that transfer data between systems, either based on a time algorithm (for example every so many minutes) or on demand (whenever a system has data to send).

Proposed SACWIS Interface Strategy

Whenever possible, states should consider using existing FAMIS to Child Support Enforcement System (IV-A/IV-D) interfaces to exchange information between SACWIS and these two systems. The Child Support Enforcement System “Locate” interface provides a robust vehicle for SACWIS to obtain identification and demographic information for a SACWIS (IV-E) participant. This interface allows the IV-D system to gather information regarding a IV-D participant from a variety of sources. Regardless of the type of interface deployed (e.g. on-line or batch), SACWIS need only send the following information for “Locate” to the IV-A system:

- SSN,
- SSN, Name,
- SSN, date of birth or,
- Name, date of birth.

The IV-A system responds with the records for every individual requested via “Locate” including: SSN, Last, first, middle name, IV-A client ID, program type, case number, residence address, date last updated, mailing address, date last updated, and client participation end date.

As an example, the “Locate” interface includes: IRS, Consumer Credit Agencies, Department of Corrections, Department of Motor Vehicles, the IV-D Electronic Parent Locate Network, and Employment Security Commission or equivalent agency, Federal Parent Locate Service, or State Election Commission or State Election Board. Many States have even more extensive interface options available through the Locate feature of their IV-D systems.

It should also be noted that the FAMIS (IV-A) system maintains the most current information regarding custodial parent and children. For non custodial parent information, the best identification/demographic information is found in the Child Support Enforcement (IV-D) System.

SACWIS States should consider enhancing the “Locate” function to permit SACWIS to generate a “Locate” request to the Child Support Enforcement (IV-D) System as though SACWIS was an integrated part of that system. As an alternative, States could provide a generalized interface that would permit SACWIS all the “Locate” functionality built into federally certified IV-D systems.

Finally, SACWIS States may find some information that is currently not exchanged between the IV-A and IV-D interfaces using the “Locate” interface, but which may be transferred during the IV-A referral process. SACWIS States should then develop a SACWIS (IV-E) to FAMIS (IV-A) interface using the “Locate” interface as the baseline and providing the particular SACWIS functional interface using a new interface “function or transaction ID” to invoke the interface.